

Effective Execution of Instruction Sequences

著者	佐藤 幸紀
号	11
学位授与番号	349
URL	http://hdl.handle.net/10097/37816

氏名（本籍地）	さとう ゆきのり 佐藤 幸紀
学 位 の 種 類	博 士（情報科学）
学 位 記 番 号	情 博 第 349 号
学位授与年月日	平成18年 3 月24日
学位授与の要件	学位規則第 4 条第 1 項該当
研 究 科、専 攻	東北大学大学院情報科学研究科（博士課程）情報基礎科学専攻
学 位 論 文 題 目	Effective Execution of Instruction Sequences（命令列の効率的実行方式に関する研究）
論 文 審 査 委 員	（主査）東北大学教授 中村 維男 東北大学教授 小林 広明 東北大学教授 青木 孝文 東北大学講師 鈴木 健一

論 文 内 容 の 要 旨

1. 結 論

半導体集積技術が向上するに伴い、マイクロプロセッサの 1 チップ上において利用可能なトランジスタの数が増加する一方で、マイクロプロセッサの電力密度が増加し設計の主要な制約となりつつある。そこで、これらの豊富なハードウェア資源を処理の高速化のために有効に活用し、かつ、処理の低消費電力化を実現する革新的な設計概念への要求が高まっている。しかし、スーパースカラ方式に代表される現在主流の単一プロセッサの設計方式は、極めて効率が悪い広域的かつ複雑なハードウェアを必要とするため、高速処理と低消費電力を両立するのが困難である。

クラスタ化アーキテクチャは、スーパースカラ方式の広域的な構造を排除した将来有望なアーキテクチャとして注目を集めている。スーパースカラ方式のデータパスを構築する単一レジスタファイルや自由度の高い演算資源間ネットワーク等の広域的な構造は、クラスタ化アーキテクチャにおいては局所化された単純な処理要素（PE）に分割される。データパスを構築する各要素において、要素のエントリ数やポート数の増加は消費電力に線形以上に影響を与えることが知られている。したがって、広域的な構造を小規模な要素に分割することは、各要素のエントリ数やポート数を削減することを可能とし、低消費電力化をもたらす。また、複雑であった構造を小規模な単純な構造に分割することは、回路の動作速度の向上にも寄与するため、処理の高速化にも有効である。しかしながら、データパスの分割を行うことにより、PE 間の通信が増加し、性能低下を引き起こす可能性がある。そこで、クラスタ化アーキテクチャにおいて高い性能を発揮するためには、プログラム中に内在する処理の局所性を踏まえて、PE 間通信が増加しないように命令をステアリング（分配）する必要がある。

本論文では、単一プロセッサにおける処理の高速化と低消費電力化の両立を目指し、クラスタ化アーキテクチャ上において命令列を効率的に実行する手法を提案する。高速処理が可能なプロセッサを実現するためには並列処理が、低電力なプロセッサを目指すためには冗長性を省いた効率的なハードウェア構成が有効である。従って、高速かつ低電力マイクロプロセッサの実現のためには、効率的に並列性を

抽出し、効率的に実行することが必要となる。そこで、プログラムの一般的な特徴の調査、性能低下の原因になりやすい命令の分類、データパスを分割する度合いとその方式の性能に与える影響、処理要素間通信の最適化、について評価を行う。

2. プログラム中の命令の並列性と逐次性

命令列を効果的に実行するためには、プログラムの一般的な特徴を把握し、その一般的な特徴を並列なハードウェアに効果的にマッピングすることが重要である。プログラム中の命令間には互いに従属あるいは独立という関係がある。互いに独立な命令は並列実行可能であるが、互いに従属の命令は、その依存関係のため、並列に実行することができない。

本章では、命令間のデータ依存関係に着目し、データ依存のあるレジスタの振舞いを解析することにより、プログラムの一般的な特徴の把握を目指す。プログラムの一般的な特徴を抽出するための尺度として、データ依存関係のある命令間の距離とレジスタファンアウトを用いる。

実行駆動シミュレーションを行い、レジスタの振舞いを解析した。その結果、データ依存関係のある命令間の距離は短いという傾向があり、命令間の距離が短くなるほど依存関係がある可能性が高いということを明らかにした。また、レジスタファンアウトは1である命令が最も多く、レジスタファンアウトが2以下の命令により全命令の9割を占めることが分かった。以降の章において、本章で解明したプログラムの一般的な特徴を概念設計へと応用することを検討する。

3. クリティカルな命令の分類

プログラムの実行時間はクリティカルパスという最長の依存のある命令列により決定される。クラスタ化アーキテクチャにおいて、命令は命令ステアリングにより各PEに分配される。命令ステアリングの結果によっては、クリティカルパス上にある命令の実行がリソース競合やPE間通信により遅れる場合がある。

本章では、データ依存関係のある命令間の距離は短いというプログラムの一般的な特徴を踏まえて、命令が処理のクリティカルパス上に存在するか否かを判断を行い、命令がクリティカルパス上に存在するか否かを統計的に分類する。命令間の距離が短いデータ依存関係のある命令は、先行する依存の結果がまだ分かっていない可能性が高いため、オペランドの状態が未解決である傾向がある。また、命令間の距離が長いデータ依存のある命令は、先行する依存の結果が既に求まっている可能性が高いため、オペランドの状態が解決済みである傾向がある。そこで、未解決オペランドを持つ命令をクリティカルパス上に存在する命令として、また、解決済オペランドを持つ命令をクリティカルパス上に存在しない命令として命令を分類する。この分類に基づき、未解決オペランドを持つ命令を優先してPEにステアリングするというクラスタ化アーキテクチャにおける命令ステアリング手法を提案する。

実行駆動シミュレーションにより、提案する命令ステアリング手法を評価した結果、従来手法と比べて性能が向上することが分かった。

4. データパス分割の効果

広域的な構造をクラスタ化により局所的な構造に分割することは、各要素内のエントリ数が削減されるため、高速かつ低消費電力な処理の実現に寄与する。しかしながら、分割の度合いを増加させるに伴い、PE間の通信が頻繁に発生し、性能低下を引き起こす可能性がある。

本章では、クラスタ化アーキテクチャにおけるレジスタファイルを分割する度合いの評価や、分割された処理要素群におけるデータ一貫性の実現方式の評価、さらに、命令発行幅を変化させた場合の評価を処理性能とレジスタファイルの消費電力の観点から行う。レジスタファイルのアクセス時間がプロセッサのサイクル時間を決定すると仮定し、CACTI モデルにより各構成におけるレジスタファイルのアクセス時間とレジスタアクセスに必要なエネルギーを算出する。実行駆動シミュレーションにより各構成方式の IPC (Instructions Per Cycle) とレジスタファイルアクセス回数を測定し、最終的にプログラムの処理速度の指標である IPS (Instructions Per Second) とプログラムを処理するために必要なレジスタファイル消費電力を見積もる。

実験の結果、データパスを分割する度合いを増加させるほど、PE 間通信のため IPC は低下するが、レジスタファイルアクセス時間は短縮され、レジスタアクセスに必要なエネルギーも削減可能になることが分かった。また、各 PE のレジスタファイルに同一の内容を保持する重複分散型のレジスタファイルより、PE 毎に個別の内容を保持する非重複分散レジスタファイルの方が高速で低消費電力なデータパスを構築可能なことを示した。以上を基に IPS とレジスタファイル消費電力の評価を行った結果、積極的に分割を行った非重複分散レジスタファイルを持つ構成が他の構成より優れていることを明らかにした。さらに、同時発行可能な命令幅を変化させた場合も同様の結果が得られ、さらに、同時発行命令幅を増加させるほど性能が向上することを確認した。

5. 隣接 PE 間の協調

本章では、積極的に分割を行った非重複分散型クラスタ化アーキテクチャにおける PE 間通信による IPC の低下を補うために、隣接する PE が協調して処理を行う方式を提案する。この隣接 PE 間の協調は、プログラムの一般的な特徴に基づいており、隣接 PE 間に 1 方向の局所的な通信経路を追加すること、隣接 PE 間協調を支援するように命令をステアリングすることから成り立つ。

性能評価実験およびレジスタファイルにおける消費電力見積もりの結果から、各 PE における演算結果を隣接 PE に直接転送する隣接 PE 間ネットワークを追加することが有効であることが分かった。さらに、レジスタファンアウトを動的に監視しながら命令をステアリングする方式が隣接 PE 間の協調効果を高めることを明らかにした。最終的に、分割を行わない構成と比べて、SPEC2000CINT ベンチマークにおいて平均 1.81 倍の速度向上を、81%少ないレジスタファイル消費電力により達成可能なことを示した。

6. 結論

本論文では、高速かつ低消費電力な処理を可能にするプロセッサアーキテクチャの実現を目指し、クラスタ化アーキテクチャにおける命令列の効率的実行方式を提案した。プログラム中の命令の依存関係を踏まえた上で命令列を構築し、それらを局所的な処理要素により実行することは、処理性能と消費電力の面で非常に効率的であることを示した。今後ますます求められている高速かつ低消費電力な単一プロセッサの実現に向けて、本論文で提案した設計指針は極めて有望であると期待される。

論文審査結果の要旨

近年、半導体加工技術の進歩によりマイクロプロセッサの高集積化が進行する一方で、消費電力が設計の主要な制約となりつつある。この背景のもと、高速かつ低消費電力のマイクロプロセッサへの要求がますます高まっている。しかし、現在主流のスーパースカラ方式に代表される方式は、大域的な構造を持つ複雑なハードウェアを必要とし、処理の局所性を活用できないため、高性能と低消費電力を両立することが困難である。本論文は、高性能と低消費電力を両立するマイクロプロセッサの概念設計として、局所化した単純な処理要素の集合から構成されるクラスタ化アーキテクチャを基盤とし、命令列を効率的に実行する手法について論じたもので、全編6章からなる。

第1章は緒論である。

第2章では、プログラムの一般的な特徴を把握するために、命令間のデータ依存関係に着目し、データ依存のあるレジスタの振舞いの解析を行なっている。その結果、プログラム中の命令列に普遍的に見られるデータ間の依存関係を命令間距離とレジスタファンアウトにより指標化している。それにより、命令列の効率的実行方式を提案し、効率的なマイクロプロセッサの基本設計の方針を示している。

第3章では、クラスタ化アーキテクチャにおいて命令を最適な処理要素に分配するために、オペランドの状態に基づく命令ステアリング手法を提案している。本手法は、プログラムの一般的な特徴を踏まえて、命令が処理のクリティカルパス上に存在するか否かの判断に基づいており、斬新な手法である。

第4章では、クラスタ化アーキテクチャにおけるレジスタファイルの分割の度合いや、分割された処理要素群におけるデータ一貫性維持の実現方式を性能と電力の観点から論じている。これらの議論は、命令列の効率的実行方式を実現するためには必要不可欠であり、非常に有益な成果である。

第5章では、クラスタ化アーキテクチャにおいて、隣接する処理要素が協調して処理を行う方式を提案している。この隣接要素間の協調は、プログラムの一般的な特徴に基づいており、局所的な通信経路を追加すること、協調を支援する命令ステアリング方式を用いることからなっている。性能評価実験から、本方式が少ないレジスタファイル消費電力で高い性能を発揮できることを示している。これは、クラスタ化アーキテクチャにおいて命令列の局所の実行を支援することは非常に効率的であることを意味しており、極めて重要な成果である。

第6章は本論文を総括し、結論としている。

以上、要するに本論文は、プログラム一般に存在する性質に着目し、次世代マイクロプロセッサの構成と概念設計について、高性能と低消費電力を実現するための重要な知見を与えたものであり、計算機設計分野ならびに情報基礎科学の発展に寄与するところが少なくない。よって、本論文は博士(情報科学)の学位論文として合格と認める。